

DevKit Memo

2017.9.29 by Alice

● Summary:

今天的進度讓 sensor 的資料收上 IoT Hub 之後可以經過 Stream analytics 儲存至 DB，並在 Powerbi 作呈現

先在跟 IoT Hub 同一個 resource group 裡建立 SQL Server 和 SQL database
並記下 Server name 跟帳號密碼

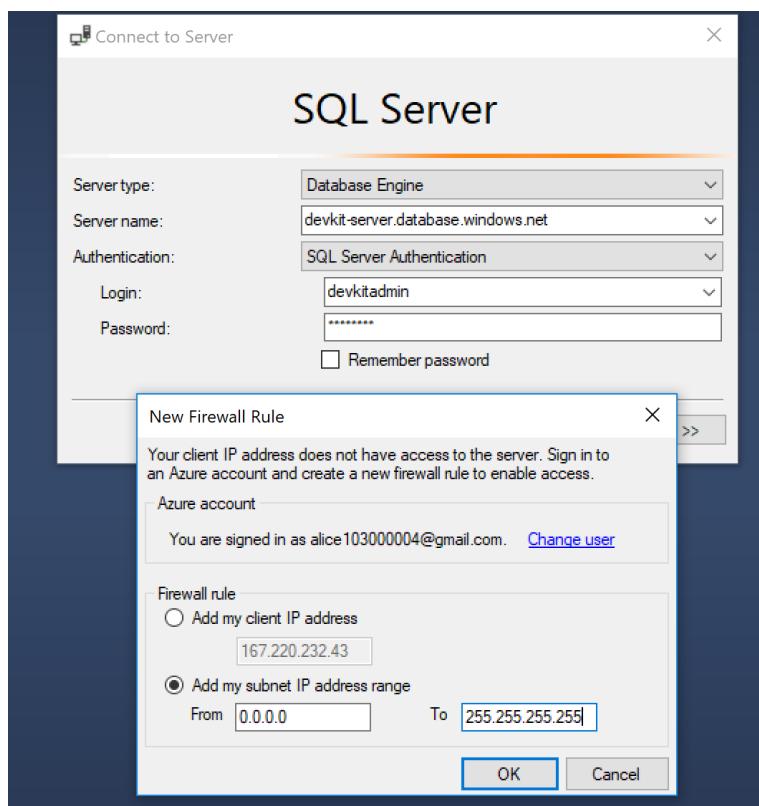
Server Name : devkit-server.database.windows.net

Account : devkitadmin

Password : Admin123



因為 devkit 使用 vs code 開發不像 Visual Studio 直接有 Server Explorer 可以查看 DB，所以我開了 SSMS 來操作 Database，下面是登入的畫面
輸入剛才記下來的 Server name 跟帳號密碼



✓ 防火牆問題:

這邊一開始我遇到 IP 被擋掉的狀況，無法登入

後來到 Azure Portal 上 SQL Server>>Firewall 新增一個 ip 區段就可以登入了，但我設的 0.0.0.0 ~ 255.255.255.255 是全開的狀態會不安全，之後會把它改回只有我的 ip 位置可以 access

devkit-server - Firewall / Virtual Networks (Preview)

SQL server

Search (Ctrl+ /)

Save Discard Add client IP

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Quick start

Firewall / Virtual Networks (Preview)

Connections from the IPs specified below provides access to all the databases in devkit-server.

Allow access to Azure services ON OFF

Client IP address 167.220.232.43

| RULE NAME | START IP | END IP | ... |
|-----------|----------|-----------------|-----|
| firewall | 0.0.0.0 | 255.255.255.255 | ... |

打開 SSMS 之後就在裡面 Create table 寫好 schema，之後就可以去 Azure 上開 Stream Analytic 把資料倒進 DB

SQLQuery10.sql - d...(devkitadmin (118)) SQLQuery9.sql - de... (devkitadmin (125))

```
CREATE TABLE [Prod].[HistoricData](
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [Temperature] [real] NULL,
    [Humidity] [real] NULL,
    [LocalTime] [datetime] NULL,
    [TemperatureAlarm] [int] NULL,
    [HumidityAlarm] [int] NULL
)
GO
```

這邊要寫一個簡單的 Query，除了將溫濕度的資料收上來，我也分別幫他們設了臨界值，一起可以將 alarm 值傳到 db 中

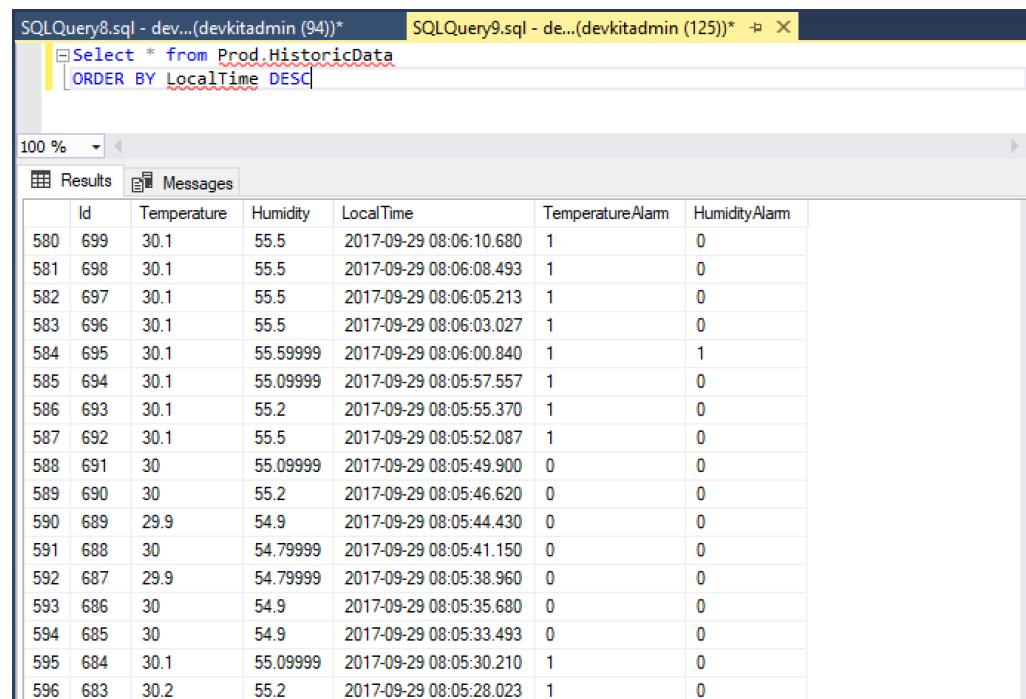
Need help with your query? Check out some of the most common Stream Analytics query patterns [here](#).

```
1 WITH HistoricData AS (
2
3     SELECT
4         Stream.[Temperature] AS Temperature,
5         Stream.[Humidity] AS Humidity,
6         CASE WHEN Stream.[Temperature] > 30 THEN 1
7         ELSE 0
8     END AS [TemperatureAlarm],
9         CASE WHEN Stream.[Humidity] > 55.5 THEN 1
10        ELSE 0
11    END AS [HumidityAlarm]
12
13    FROM
14        [newhub] stream
15
16 )
17 SELECT * INTO [newdb] FROM HistoricData
18
```

因為我原本建的 table 中沒有 temperatureAlarm 跟 humidityAlarm 這兩個 column
所以我後來又再用 SQL 語法把這兩個 column 加到 TABLE 裡

```
SQLQuery8.sql - dev...(devkitadmin (94))* ➔ X SQLQuery4.sql - dev...(devkitadmin (98))* ➔ SQLQuery1.sql - dev...(devkitadmin (122))*
ALTER TABLE Prod.HistoricData ADD TemperatureAlarm INT NULL, HumidityAlarm INT NULL ;
```

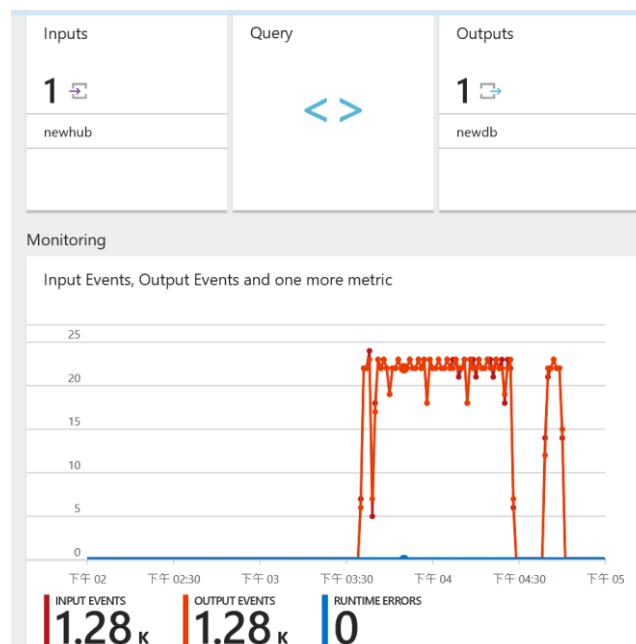
整個把環境弄好之後就可以讓 DevKit 把資料收上來了
這邊可以從 SSMS 去觀察資料進 DB 的狀況 如下圖:



The screenshot shows a Microsoft SQL Server Management Studio (SSMS) interface. At the top, there are three tabs: 'SQLQuery8.sql - dev...(devkitadmin (94))*', 'SQLQuery9.sql - dev...(devkitadmin (125))*', and 'SQLQuery1.sql - dev...(devkitadmin (122))*'. The middle tab is active. Below the tabs, a query is displayed: 'Select * from Prod.HistoricData ORDER BY LocalTime DESC'. The results grid shows data from the 'Prod.HistoricData' table. The columns are: Id, Temperature, Humidity, LocalTime, TemperatureAlarm, and HumidityAlarm. The data consists of approximately 50 rows of sensor readings. The 'LocalTime' column shows dates and times from September 29, 2017, at 08:06 to 08:05.

| 580 | 699 | 30.1 | 55.5 | 2017-09-29 08:06:10.680 | 1 |
|-----|-----|------|----------|-------------------------|---|
| 581 | 698 | 30.1 | 55.5 | 2017-09-29 08:06:08.493 | 1 |
| 582 | 697 | 30.1 | 55.5 | 2017-09-29 08:06:05.213 | 1 |
| 583 | 696 | 30.1 | 55.5 | 2017-09-29 08:06:03.027 | 1 |
| 584 | 695 | 30.1 | 55.59999 | 2017-09-29 08:06:00.840 | 1 |
| 585 | 694 | 30.1 | 55.09999 | 2017-09-29 08:05:57.557 | 1 |
| 586 | 693 | 30.1 | 55.2 | 2017-09-29 08:05:55.370 | 1 |
| 587 | 692 | 30.1 | 55.5 | 2017-09-29 08:05:52.087 | 1 |
| 588 | 691 | 30 | 55.09999 | 2017-09-29 08:05:49.900 | 0 |
| 589 | 690 | 30 | 55.2 | 2017-09-29 08:05:46.620 | 0 |
| 590 | 689 | 29.9 | 54.9 | 2017-09-29 08:05:44.430 | 0 |
| 591 | 688 | 30 | 54.79999 | 2017-09-29 08:05:41.150 | 0 |
| 592 | 687 | 29.9 | 54.79999 | 2017-09-29 08:05:38.960 | 0 |
| 593 | 686 | 30 | 54.9 | 2017-09-29 08:05:35.680 | 0 |
| 594 | 685 | 30 | 54.9 | 2017-09-29 08:05:33.493 | 0 |
| 595 | 684 | 30.1 | 55.09999 | 2017-09-29 08:05:30.210 | 1 |
| 596 | 683 | 30.2 | 55.2 | 2017-09-29 08:05:28.023 | 1 |

下圖是 Stream Analytics input output 的狀況:



成功存到 db 後就可以拉到 PowerBI 了！

MXChip DevKit Sensor

